

On implementation of the minimal perimeter triangle enclosing a convex polygon algorithm

Victor Ermolaev

June 8, 2016

Abstract

A high-level description of an algorithm which computes the minimum perimeter triangle enclosing a convex polygon in linear time exists in the literature. Besides that an implementation of the algorithm is given in the subsequent work. However, that implementation is incomplete and omits certain common cases. This note's contribution consist of a modern treatment of the algorithm and its consequent simple and robust implementation.

1 Introduction

A very simple and ingenious algorithm of finding a minimal perimeter triangle enclosing a convex polygon was given by Bhattacharya and Mukhopadhyay [2002]. The algorithm has a linear complexity w.r.t. the number of edges of the polygon. This makes it very attractive for applications.

Let a convex polygon be given, such that no two neighbouring edges are colinear. The algorithm suggests to treat every edge sequentially as one of the sides of the optimal triangle; the second (suboptimal) side is bootstrapped and one optimization step is performed. For the exposition of the algorithm suppose that two sides of the minimal perimeter enclosing triangle are found and form a wedge. A quest for the third side is an optimization procedure w.r.t. edges and vertexes of the polygon: the third side will either be colinear to one of the edges or pass through one of the vertexes. Findings in [Bhattacharya and Mukhopadhyay, 2002] state that:

- if the wedge and extension of a polygon edge form a triangle, then the third side will be colinear to the edge if a circle escribed to the triangle touches the edge (and not its extension);
- if a circle is inscribed into the wedge such that that it passes through one of the polygon vertexes, then the third side will be a tangent to the circle at the vertex provided it is also tangent to the polygon.

Finding such a side embodies one optimization step of “closing” the wedge in the optimal way. The next optimization step is performed for a wedge formed by the first side and the newly found side. Such wedge “flipping” continues until no further improvement in perimeter is possible.

As described, the algorithm relies on four subsidiary procedures of fitting a circle into a wedge while satisfying some constraints. The first two sub-problems allow us to bootstrap the algorithm, whereas the last two constitute one optimization step. For the bootstrapping part, call a wedge degenerate, if its arms are parallel.

Subsidiary problems

1. Given a degenerate wedge and a line crossing it, find a circle touching both of the wedge’s arms and the line;
2. given a degenerate wedge and a point between its arms, find a circle touching the arms of the wedge and passing through the point;
3. given a wedge and a line crossing it, find an excircle for the generated triangle such it touches the given line;
4. given a wedge and a point between its arms, find a circle inscribed into the wedge and passing through the point.

Medvedeva and Mukhopadhyay [2003] sketch an implementation which is further developed in [Medvedeva, 2003]. That implementation relies on slope-intercept representation of lines, which leads to the numerous case distinction. We develop an alternative treatment of the sub-problems employing simple linear algebra techniques. Our approach eliminates the infinite-slope problem while leaving the main body of the algorithm intact. Our proposition reduces the overall complexity of the algorithm leading to a simpler and more direct implementation, for example, on top of a linear algebra library. In particular, we provide a reference to our implementation hosted on GitHub [Ermolaev, 2016]. The GitHub project includes sample polygons from [Pârvu and Gilbert, 2014] as test cases.

2 Linear algebra approach to sub-problems

With $l(t)$ we denote the parametric form of a line defined by two distinct point P and Q , $l(t) = (Q - P)t + P$. With $w(S_0E_0, S_1E_1)$ we denote a wedge with arms S_0E_0 and S_1E_1 . The wedge is called degenerate if its arms are parallel. Let 2D cross product of two vectors be defined as the signed area that these vectors span.

2.1 Degenerate wedge and line

Let a degenerate wedge $w(S_0E_0, S_1E_1)$ and a line defined by points A and B be given. This case is depicted in Figure 1. Assume that the line is not parallel to the arms and the half of the distance between the arms equals to r . Obviously, inscribed circles will have their centres I_c lying on a line parallel to the arms, such line will be parametrized as, for example, $l(t) = (E_1 - S_1)t + I$, where I is any point equally distanced from the arms. Radii of the circles will equal to r . To find the centres I_c we consider the line-to-point distance equation:

$$\frac{(B - A) \times (A - I_c)}{\|B - A\|} = \pm r. \quad (1)$$

After having plugged in the parametric representation of I_c , we solve it for t

$$t_{1,2} = \frac{(B - A) \times (A - I) \pm \|B - A\| r}{(B - A) \times (E_1 - S_1)} \quad (2)$$

2.2 Degenerate wedge and point

Let a degenerate wedge $w(S_0E_0, S_1E_1)$ and a point P between its arms be given. This case is depicted in Figure 2. As before, assume that the half of the distance between the arms is r . Similarly, inscribed circles will have their centres lying on a line parallel to the arms — $l(t) = (E_1 - S_1)t + I$, where I is any point equally distanced from the arms. Radii of the circles will equal to r . The centres I_c satisfy the point-to-point distance equation:

$$\|P - I_c\|^2 = r^2 \quad (3)$$

Substituting I_c with the parametric representation of $l(t)$ brings us to

$$\|E_1 - S_1\|^2 t^2 + 2(I - P) \cdot (E_1 - S_1)t + \|I - P\|^2 - r^2 = 0 \quad (4)$$

It is not difficult to see that this equation always has two roots corresponding to different centres of the inscribed circles.

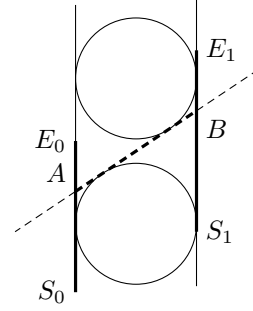


Figure 1: Bootstrap step

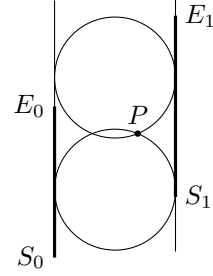


Figure 2: Bootstrap step

2.3 Wedge and line

In Figure 3 a wedge $w(CA, CB)$ and a line l (dashed) forming $\triangle ABC$ are depicted. We look for an excircle with centre $I = (x_I, y_I)$ and radius r for $\triangle ABC$ tangent to AB . Let s be a semiperimeter of $\triangle ABC$, then $r = \sqrt{\frac{s(s-a)(s-b)}{s-c}}$. Let $(x_{AB}, y_{AB}) = B - A$ and $(x_{AC}, y_{AC}) = C - A$. The distance equations between AB and I and AC and I read

$$\frac{(B - A) \times (A - I)}{c} = \pm r \quad (5)$$

$$\frac{(C - A) \times (C - I)}{a} = \pm r \quad (6)$$

leading to the following systems of linear equations

$$\begin{pmatrix} -y_{AB} & x_{AB} \\ -y_{AC} & x_{AC} \end{pmatrix} \begin{pmatrix} x_I \\ y_I \end{pmatrix} = \begin{pmatrix} B \times A \pm rc \\ C \times A \pm ra \end{pmatrix} \quad (7)$$

Solution of this system always exists, unless l is parallel to one of the arms of the wedge. The correct I has to satisfy

$$\| |(C - B) \times (B - I)| - rb \| = 0. \quad (8)$$

In Figure 3 all solutions are plotted and the right excircle is given in thick.

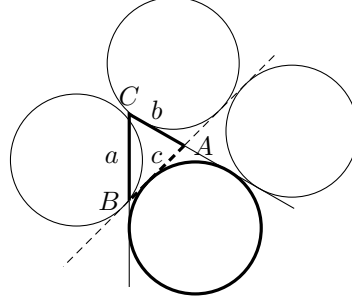


Figure 3: Typical example where case distinction for [Medvedeva and Mukhopadhyay, 2003] is required

2.4 Wedge and point

It is known that given a wedge $w(CA, CB)$ and a point P within it, centres of inscribed circles will lie on bisector of the wedge. The bisector is defined with points C and $D = (A - B) \frac{a}{a+b} + B$ and thus parametrized with $l(t) = (D - C)t + C$. To find the centres I of the inscribed circles we solve

$$\|I - P\|^2 = \left\| \frac{(A - C) \times (C - I)}{b} \right\|^2 \quad (9)$$

for t . Plugging in the parametric representation of the bisector produces a quadratic equation w.r.t. t

$$\left(\|D - C\|^2 - \left(\frac{(A - C) \times (D - C)}{b} \right)^2 \right) t^2 + 2(D - C) \cdot \left(\frac{(A - C) \times (D - C)}{b} \right) t + \|C - P\|^2 = 0 \quad (10)$$

Simple analysis shows, that this equation always has two roots which correspond to the centres of a smaller and a larger inscribed circles, see Figure 4. According to [Bhattacharya and Mukhopadhyay, 2002] we are interested in the larger circle.

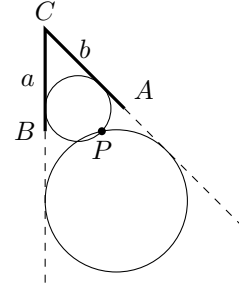


Figure 4: Typical example where case distinction for [Medvedeva and Mukhopadhyay, 2003] is required

3 Conclusion

The proposed line representation alleviates the fitting procedures removing the separate treatment for border cases. The parametric representation of lines guarantees existence of solutions of subsidiary problems. The algorithm was implemented in TypeScript [Microsoft, 2012] for use in browsers after transpilation to JavaScript and available on GitHub [Ermolaev, 2016].

References

- Binay Bhattacharya and Asish Mukhopadhyay. On the minimum perimeter triangle enclosing a convex polygon. In *Discrete and Computational Geometry*, pages 84–96. Springer, 2002.
- Victor Ermolaev. An implementation for web of the minimal perimeter enclosing triangle algorithm, 2016. URL <https://github.com/quicky84/minimal-perimeter-triangle>.
- Anna Medvedeva and Asish Mukhopadhyay. An implementation of a linear time algorithm for computing the minimum perimeter triangle enclosing a convex polygon. In *CCCG*, volume 3, pages 25–28, 2003.
- Anna Valentinovna Medvedeva. Computing the minimum perimeter triangle enclosing a convex polygon: Theory and implementation. Master’s thesis, University of Windsor, 2003.
- Microsoft. Typescript, 2012. URL <http://www.typescriptlang.org/>.
- Ovidiu Pârvu and David Gilbert. Implementation of linear minimum area enclosing triangle algorithm. *Computational and Applied Mathematics*, pages 1–16, 2014.